

1. Realice un programa que muestre por pantalla el texto "Hola mundo!" sin comillas.
2. ¿Algún número mayor que 10? Lee por la entrada estándar (normalmente teclado) tres números usando `prompt()`. El comando `prompt()` te permite leer una línea completa. Si al menos uno de los valores ingresados es mayor a 10, mostrar por la salida estándar (normalmente el teclado) "ALGUNO MAYOR QUE 10". En caso contrario mostrar "NINGUNO MAYOR QUE 10".
3. Lee por la entrada estándar (normalmente teclado) día y mes en formato numérico, cada uno en una línea. Debes realizar un programa que imprima SI si es navidad (el 25 del 12) y NO si es no es navidad.
4. Aumento de sueldo De un operario se obtiene por teclado una línea con su sueldo (sin decimales) y en otro línea los años de antigüedad. Se pide un programa que lea los datos de entrada e informe: a) Si el sueldo es inferior a 500 y su antigüedad es igual o superior a 10 años, triplicar el sueldo. b) Si el sueldo es inferior a 500 pero su antigüedad es menor a 10 años, doblar sueldo. c) Si el sueldo es mayor o igual a 500 mostrar sin cambios. Imprime el sueldo resultante.
5. Escribe un programa que lea por entrada estándar una calificación numérica entre 0 y 10 (con decimales) y la transforma en calificación alfabética según la siguiente tabla.
 - de 0 a <3 Muy deficiente
 - de 3 a <5 Insuficiente
 - de 5 a <6 Bien
 - de 6 a <9 Notable
 - de 9 a 10 Sobresaliente
6. Escribe un programa que lea en una línea el número de horas trabajadas y en otra línea el precio por hora. Calcula el salario neto semanal de un trabajador en función del número de horas trabajadas y la tasa de impuestos de acuerdo a las siguientes reglas:
 - Las primeras 35 horas se pagan a tarifa normal.
 - Las horas que pasen de 35 se pagan a 1.5 veces la tarifa normal.Al sueldo bruto se le aplicarán las siguientes tasas de impuestos:
 - Los primeros 500 euros son libres de impuestos.
 - Los siguientes 400 tienen un 25% de impuestos.
 - Los restantes un 45% de impuestos.
7. Realiza un programa que lea un numero N y muestre por la salida estándar todos los pares desde 2 hasta N (inclusive). Cada número se mostrará en una línea distinta. N siempre será mayor o igual a 2.
8. Realiza un programa que lea un numero N y muestre por la salida estándar todos los impares desde N(inclusive) hasta 1 . Cada número se mostrará en una línea distinta. N siempre será mayor o igual a 1.
9. Desarrollar un programa que muestre la tabla de multiplicar del numero X, hasta el valor Y. Ambos los leerá de la entrada estándar y estarán en una sola línea separados por un espacio, siendo $-50 \leq X \leq 50$, $1 \leq Y \leq 20$ La tabla la mostrará con una entrada por línea. Entre cada elemento habrá un único espacio de separación.
10. Realiza un programa que lea una secuencia de notas (con valores que van de 0 a 10, sin decimales). Se leerá una nota por línea. La entrada acabará con el valor -1. El programa debe mostrar por salida estándar "SI" si hay algún 10, "NO" en caso contrario.

11. Realiza un programa que lea una cantidad de dinero (múltiplo de 5) y tras ello desglose el cambio en billetes de 500,200,100,50,20,10,5 intentado dar el mínimo número de billetes. El programa mostrará por la salida estándar el número mínimo de billetes.
12. Escribe una función que reciba dos enteros y devuelva su suma. La función no debe leer nada de la entrada estándar ni mandar ningún dato por la salida estándar. La función debe definirse con este estilo: `function suma(a, b)`
13. Realiza una función que me devuelva si un número es o no bisiesto. Un número es bisiesto si es divisible entre 4 y entre 400. **function esBisiesto(anyo)**
14. Escribe una función que reciba dos decimales con la base y altura de un rectángulo y devuelva su perímetro. La función no debe leer nada de la entrada estándar ni mandar ningún dato por la salida estándar.

Nota: se devolverá un decimal (float). El juez internamente se encargará de redondearlo a 2 decimales. La función debe definirse con este estilo: **function perimetroRectangulo(a, b)**

Para limitar los decimales solo tenemos que usar la función `toFixed()`

`valor=numero.toFixed(2)`